Simplifying Bibliometric Analysis with Python Streamlit

Leticia Naomi Asano[1] **, Encarna Sosa Sánchez**[2] **, Lucineia Heloisa Thom**[1]

[1] Institute of Informatics, Federal University of Rio Grande do Sul, Porto Alegre, Brazil
[2] University of Extremadura, Department of Computer Systems and Telematics Engineering, Cáceres, Spain

leticia.asano@gmail.com.br, esosa@unex.es, lucineia@inf.ufrgs.br

Abstract. ***Research Context:*** *Bibliometric analysis is a research methodology that has become increasingly popular in recent years, in different domains of science. Many tools have been developed to assist researchers in making sense of this data.* ***Scientific and Practical Problem:*** *The tools available to perform bibliometric analysis today are mostly integrated only with Web of Science (WoS) and SCOPUS, limiting researchers who wish to diversify their sources. They do not offer any support in data preprocessing. Many scientists choose to use more than one tool to adapt to their needs, as rarely is one tool complete.* ***Proposed Analysis:*** *After studying the available software, packages, and articles on bibliometric analysis, we propose a new tool that can be adapted to multiple science databases, performs bibliometric analysis, supports Excel to facilitate data preprocessing, and is designed to be both user-friendly and accessible to a broad audience.* ***Related IS Theory:*** *Related IS Theory fields are Organizational knowledge creation and Information processing theory. Diffusion of innovations theory is also relevant.* ***Research Method:*** *Design Science Research (DSR) was applied to identify the necessities for bibliometric analysis not covered by the available tools and identify how we can improve these gaps in an efficient manner.* ***Summary of Results:*** *The resulting artifact can be a tool for scholars of various fields. The direct integration with Excel facilitates the preprocessing and integration with many databases. It facilitates the extraction of missing metadata and features a simple, intuitive interface.* ***Contributions and Impact to IS area:*** *This study proposes a novel approach to performing bibliometric analysis, utilizing multiple science databases and integrating all necessary visualization tools.*

1. Introduction

Bibliographic analysis is an efficient research methodology that enables practitioners to gather important network data across a field of science, yielding quantitative results [Donthu et al. 2021]. However, the tools to assist this analysis today only integrate with certain databases of scientific research - traditionally SCOPUS and WoS. Additionally, they often include a learning curve as features may not be so intuitive and are complex.

[Donthu et al. 2021] shows that the growth in Bibliographic Analysis from 2005 to 2020 in the business industry was significant. We can still find a recent number of references that chose this methodology in the field of computer science ([Koçak and Akçalı 2025],

[Mioto and Vignatti 2025], [Nolêto et al. 2023], [Uzeda et al. 2023]). When checking the references, we see that most researchers ([Koçak and Akçalı 2025], [Nolêto et al. 2023], [Uzeda et al. 2023]) chose to work with specific databases like WoS and SCOPUS, because those databases offer bibliometric tools and article metadata that enable the analysis. Although these platforms present a vast amount of content in various fields of science, only using the same sources can exclude important sources from the research work.

When the researcher wants to work with more than one database, it is often necessary to use an API to retrieve the data ([Mioto and Vignatti 2025]). This approach works with the downside that it is not too accessible to the other members of the research community who are not familiar with computer science and may be discouraged from implementing code to obtain data.

As for the tools that are already available, we see that most of them also only integrate with WoS and SCOPUS. A review of the tools by [Moral-Muñoz et al. 2020] reveals that software and packages available today are not complete, and the ones that are still do not support integration with every database. This makes it difficult for researchers who want to work with bases that are more specific to their research areas.

A tool that is proven to be accessible is VosViewer, which is often chosen by scholars ([Uzeda et al. 2023], [Koçak and Akçalı 2025]). A practitioner who wishes to use this tool with alternative databases has to do some preprocessing of the data to match one of its allowed formats, leading to possible data loss. Another tool that is very complete, accordingly to the review of [Moral-Muñoz et al. 2020], is the Bibliometrix package, which can also be used with the visualization software Biblioshiny. It also only integrates with SCOPUS, WoS, and Dimensions, and contrary to VosViewer, does not accept any other formats like RIS, Open Citation, or Semantic Scholar.

Based on the difficulty of selecting a tool to work with alternative databases when performing bibliographic analysis, we developed an approach using the Crossref API ([Crossref 2025]) and the Python Streamlit framework ([Streamlit 2025]). This tool accepts any Excel file (comma-separated values format) that contains the required fields (further details in section 3), which is a traditional way scholars process their references. The main advantage of working with Excel is that not only can it be easily integrated with citation management tools for data preprocessing, but it also supports joining together data from multiple sources. The source code for the tool can be obtained here `https://github.com/bpm-researcher/bibliographic-analysis-tool.git`. The streamlit app can be accessed here `https://bibliographic-analysis-tool.streamlit.app/`.

2. Theoretical Reference

To perform bibliometric analysis, scholars usually perform certain steps such as extracting data, processing data, finally analyzing and visualizing the results. These steps may usually involve many different computational tools, for example scripts to extract data, software such as Excel and Zotero to clean the references and resources to build graphs and display it in a comprehensive way. In this section we will go through these steps in further details, and discuss similar articles that also explore these factors to understand the needs of the scientific comunity in terms of a tool for simplifying bibliometric analysis.

2.1. Bibliometric Analysis

The Bibliometric Analysis methodology is useful for processing a large amount of data. Its methods can be used to observe trends and gaps in a research field, identify where authors collaborate more and less, and the points where the research has yet to reach [Donthu et al. 2021]. By taking numerous metadata into consideration, scientists can use mathematics and statistics to draw objective conclusions regarding the current status of their field of research. Many of such analysis have been done in computer science in the course of the years ([Neely 2005], [Maz-Machado et al. 2010], [Ellegaard and Wallin 2015], [Kheddar 2025]) and they are directed to many uses such a research can have: identifying trends, verifying gaps, understand the scope of a determined field and forming new ideas.[Donthu et al. 2021] summarizes the reasoning behind this choice of analysis when starting research. According to them, bibliometric analysis is useful to make sense of the numerous metadata and use this to understand the state of a field of research. They argue that this use is recommended when we have a comprehensive scope and a large amount of data that hinders manual review. Another reason why scholars chose this method of research is the possibility to analyze the results quantitatively, avoid bias, and get a more objective view.

This type of analysis may be linked to a Systematic Literature Review (SRL) or a meta-analysis by a scholar who wishes to determine a methodology for their research [Passas 2024]. At first glance, bibliometric analysis may seem similar to a meta-analysis because of its quantitative nature. [Passas 2024] argues that those two types of analysis have distinct objectives, as the first focuses on the structure of the target field and the relationship between its research constituents, while the second leans towards studying the relations between different variables. Meta-analysis often requires careful consideration of the jeopardy of bias ([Graeff et al. 2023], [Alskaf et al. 2022]). In fact, such studies are more subject to this danger because results depend directly on the primary studies, and thus are affected by their quality and the way they are selected. We observe that favorable results have a higher tendency of being selected, which increases bias for this type of work. Another factor to take into consideration when performing meta-analysis is the fact that different studies may differ in measures, populations, interventions or quality; therefore, the practitioner has to carefully work with heterogeneous data to acquire consistent results.

A SLR is a popular methodology ([Tenório et al. 2021], [de Sousa Araújo et al. 2025], [Kudo et al. 2022]) indicated when there is a smaller number of primary studies and a more narrow scope. SRLs involve a manual processing of all articles, and thus, it is necessary to work with a more niche field to obtain quality results. Because of this, it tends to construct a more qualitative analysis, which can be subjected to bias from the author, who tends to focus more on the factors that are favorable to their previous vision on the issue.

It is not unusual to see studies that perform SLR and later meta-analysis ([Graeff et al. 2023], [Alskaf et al. 2022]) to mitigate the bias of the research and add a quantitative perspective to the results. [Costa et al. 2023] argues the benefits of integrating SLR with bibliometric analysis to harvest the benefits of having both qualitative and quantitative analysis. Overall, the choice of a review is dependent on the objectives of the researcher, each of them having its advantages and drawbacks. It is important to choose accordingly to the objectives of the research and characteristics of the field of study.

When a researcher has chosen that a bibliometric analysis is more suitable for their research objectives and dataset, there is a set of techniques that they can use in this process. We often associate bibliometric analysis with performance analysis and science mapping. We will detail those strategies over the next two sessions.

2.2. Performance Analysis

When performing bibliometric analysis, we usually want to obtain data on the size of the field. One of the ways to achieve this is through performance analysis, where we count citations of papers in a certain way to make conclusions on the field of research [van Raan 2014]. Note that the citation data of the papers must be obtained so that we can calculate various relevant metrics for the research constituents (article, author, publisher, subject, and so on). Some indicators for performance analysis are the total number of citations, h-index, g-index, the Lorenz curve and the Gini coefficient, which we will explain in further details in the following paragraphs.

Many types of metrics for performance measures exist. Usually, researchers are interested in the total number of citations to get an idea of the size of the field. It is also interesting to calculate this number per article or author, to understand if there are research constituents that have citations way above or below the average of citations, thus revealing an unbalanced importance given to their contribution. It is often popular to use h-index and g-index to have a more strategic vision of the performance of the research constituent.

It is common to pair the previously cited metrics with the Lorenz curve and the Gini coefficient, which are related measures of cumulation. The Lorenz curve L(p), where p is a fraction of a resource (in this case, p would be in reference to the number of citations for the research constituents), can be defined as the cumulative proportion of the total resource held by the bottom p fraction of the population. Thus if F(x) is a cumulative distribution function with a mean μ (with F(x) having continuous derivatives and the mean $\mu = \int x\, dF(x)$ finite), then we can write:

$$L(p) = \mu^{-1} \int_0^p F^{-1}(x) \tag{1}$$

This definition can be found in further details in [Gastwirth 1971]. The important point of the Lorenz curve for our discussion is that the Lorenz curve can be used to easily analyze the cumulation of citations per research constituent. Indeed, the proximity of the Lorenz Curve to the identity line serves as a useful guideline to verify if the distribution is balanced among constituents (curve closer to the identity line) or if it is less equally distributed (curve closer to the x-axis).

A complementary information to the Lorenz curve that may be of interest is the Gini coefficient, which can be defined as the area representing the difference between the studied Lorenz curve and the equality. In our context, equality is the identity line, and thus we can define the Gini Coefficient as follows ([Dorfman 1979], [Farris 2010]):

$$G = 2 \int_0^1 [p - L(p)] dp \tag{2}$$

Demonstrating this is out of the scope of this work and can be checked in further detail in the literature. The important thing to focus on in this work is that the Gini

coefficient is a number between 0 and 1, and so a researcher may interpret that a coefficient closer to 1 refers to a more equally distributed dataset, whereas an almost null result demonstrates that the citations are much more concentrated between a few constituents.

Here we cited the main indicators that are commonly used in performance analysis. This type of analysis is more commonly found in most bibliometric reviews, and provides important insights into the size of contribution of each research constituent. A more comprehensive list of the metrics can be found in [Donthu et al. 2021] and in the literature. In the next section, we further discuss attributes of scientific mapping.

2.3. Science Mapping

At this phase of the bibliographic analysis, the practitioners are more interested in the way the research constituents relate to each other. Science Mapping (SM) has tools to cluster research constituents in order to obtain trends of research and possible gaps, providing insights into how the information is being communicated among the members of the dataset. To this end, tools such as co-citation analysis and bibliographic coupling are used together with algorithms to build thematic clusters, and such techniques are augmented with the assistance of network analysis metrics [Donthu et al. 2021].

Co-citation analysis is often used ([Pan et al. 2019], [Modak et al. 2025]) to obtain thematic clusters. They are constructed by mapping the research constituents that are cited together many times in the dataset, and using algorithms to group them and obtain information on the dataset. It is important to notice that this takes into account the number of recurrences of the publication, leading to the prevalence of older contributions that tend to be more cited ([Donthu et al. 2021]).

Similarly, bibliographic coupling can also be used to construct clusters ([Modak et al. 2025]). In this case, the logic is to group works that have shared references and analyze clusters formed in this way. Consequently, this enables the study of newer fields of research, since this does not rely on the number of citations like the previous method, and can be used by practitioners to get a better sense of the present state of research.

To build clusters efficiently, a few algorithms are popular in the literature, such as Hierarchical Clustering [Nielsen 2016], Lovain Method [Blondel et al. 2008], Simple Centers Algorithm [Hakimi 1964], Clauset-Newman-Moore Greedy Modularity [Clauset et al. 2004] and Label Propagation [Cordasco and Gargano 2011]. Since Hierarchical Clustering is less scalable for large amounts of data ([Monath et al. 2021]) and the Simple Centers Algorithm requires that a k-number of clusters is pre-defined by the user, making the results limited by this user suggestion ([Ikotun et al. 2023]), we will focus here on the remaining three algorithms.

The Louvain method is widely cited in various contexts ([Laurett and Ribeiro 2022], [Barbon Jr et al. 2017], [Lemos et al. 2024]). It is a heuristic algorithm guided by modularity optimization and it is originally built to process large amounts of data ([Blondel et al. 2008]) faster than similar algorithms proposed. Similarly, the Greedy algorithm in the remainder of this work, functions similarly to the Louvain algorithm, except it tries to maximize the increase in modularity as much as possible. It is proposed by [Clauset et al. 2004] and works very well with medium communities.

Lastly, the Label Propagation algorithm works a bit differently. It tries to find communities by propagation of labels and was first introduced by [Cordasco and Gargano 2011]. This algorithm is not reliable, meaning it can find different communities each time it is run.

To contribute to the analysis that can be made by the clusters, specific network metrics can be used to take more sensitive insights on each of the research constituents, such as betweenness centrality, eigenvector centrality and closeness centrality. If u is a research constituent, then its betweenness centrality B(u) can be expressed by:

$$B(u) = \sum \frac{\delta_{v,w}(u)}{\delta_{v,w}} \tag{3}$$

where $\delta_{v,w}(u)$ is the number of shortest paths that pass through the node u, and $\delta_{v,w}$ represents the total number of shortest paths for the entire network ([Brandes 2001]). This measure exposes nodes that act as bridges to other nodes within the network.

The eingenvector centrality of research constituent u_i can be calculated by the following:

$$u_i = \epsilon^{-1} \sum_{j=1}^{n} A_{ij} u_j, \tag{4}$$

as explained by [Negre et al. 2018], where ϵ^{-1} is a constant. Accordingly, the eigenvector is thus defined by the sum of the centrality of all constituents that pass through u_i and an edge A_{ij}. For practical reasons, a node with a high value in eigenvector centrality can be understood as a node connected to other important nodes in the network [Donthu et al. 2021]

Page rank is an example of a eigenvector centrality that is used to classify the importance of a node in a rank of pages ([Ding 2009]). This measure is sometimes also applied to clustering to classify the nodes. A formula for PageRank is

$$\bar{R}_n = \frac{(1-d)}{N} + d\left(\frac{PR(T_1)}{C(T_1)} + \frac{PR(T_2)}{C(T_2)} + \cdots + \frac{PR(T_n)}{C(T_n)}\right) \tag{5}$$

Lastly, the closeness centrality is also based on the shortest paths of a constituent u and measures how close this node is to all other nodes in the network. Thus, closeness centrality is measured, accordingly to [Brandes 2001], as follows:

$$C(u) = \frac{1}{\sum_v \delta_{v,w}} \tag{6}$$

A simple metric that can also be taken under consideration is simply the Degree, which refers generally to the number of connections of a node inside the network ([Donthu et al. 2021]). These metrics can be used to measure the importance of nodes within a network. For example, special attention can be given to a node with higher betweenness centrality, as it may indicate a focal point of the research. Eigenvector centrality can be used to identify seminal constituents of the network, while closeness centrality can be used to gain insight into the foundational papers.

Another tool for science mapping is co-word analysis. Researchers build word clouds ([Soares et al. 2019]) to understand which concepts are related to an idea they

consider key in the context of their research. These ideas may be taken from the keywords, titles or abstracts. Authors may also use it to understand research trends, for example, by seeing what words are associated with the word future.

2.4. Related Articles

Bibliometric Analysis is a methodology that is amply used in computer science (and other disciplines), as it enables the processing of a large amount of data and quantitative results for analysis. A recent example is [Koçak and Akçalı 2025], where over 4000 articles were studied using the metadata extraction from the Web of Science (WoS). This database offers immediate extractions of the metadata of hundreds of articles at a time, including crucial information on citation and reference, data without which most bibliometric analysis would be incomplete. This facilitation of the WoS platform is not repeated on other platforms. [Koçak and Akçalı 2025] itself, which is an article of the medical industry, mentions that other reference databases from the field, such as PubMed were not used in this research, due to the facility of using WoS for extracting this data. In the field of computer science, ACM Digital Library and IEEExplore are platforms often used to gather initial data, but currently, neither of them offers options to export citations as easily as WoS, and their imports do not include citation and reference data.

Additionally, articles use tools that integrate better with WoS. [Nolêto et al. 2023] uses the bibliometrix package, which is an R tool constructed to perform bibliometric analysis. While it offers many features, it only integrates with databases extracted from WoS. [Koçak and Akçalı 2025] and [Uzeda et al. 2023] use vosViewer, a common tool for bibliometric analysis that integrates with WoS and Scopus - another platform that currently facilitates the extraction of metadata, such as WoS.

WoS is a rich database with numerous relevant papers, but there are downsides to limiting the search to only one database. Due to rigorous indexing criteria, many relevant journals may not be included and pre-printed articles are excluded from the research. This may result in incomplete and outdated conclusions. Additionally, the nets of collaboration that we usually investigate in bibliometric analysis are restricted to only the networks that are allowed in the WoS database.

To deal with the difficulties of gathering complete metadata from multiple bases, researchers chose to use APIS, such as [Mioto and Vignatti 2025]. Using an API provides more control over which data can be retrieved, given that this data is available. In this case, the data was obtained in CSV format and later had to be processed to be visualized in Gephi, which is another visualization tool. Usually, to use gephi it is necessary to construct a GEXF file, which is commonly done using scripts. The approach of [Mioto and Vignatti 2025] (constructing an API and building a GEXF) is not very approachable for researchers outside the computer science community, since a lot of coding is necessary to perform the processing of data.

In [Moral-Muñoz et al. 2020] we see a review of tools for bibliometric analysis. The review was performed by testing software for performance analysis and for science mapping analysis and resulted in the testing of 11 tools and 6 libraries. The results show that all of the tools support either SCOPUS or WoS, but do not support all the tested databases. The tools that offer a scientific mapping feature provide, accordingly to [Moral-Muñoz et al. 2020], interesting visualization features, but do not offer

pre-processing of data. The most complete tool seems to be Bibliometrix (or its visualization tool biblioshiny), but it integrates with SCOPUS, WoS and Dimensions only, which makes it difficult to process data in order to use the software if the practitioner wishes to use multiple databases.

In the meantime, streamlit's popularity as a Python library for easily creating user interfaces to share scientific results has grown, as researchers use this technology to share research results ([Muhammad Qadir et al. 2025], [Guleria et al. 2023]). In [Dol and Jawandhiya 2024], Streamlit's power is cited to enable visualization of data for educational purposes. The simplicity of using this library and the analytical power of Python libraries suggest that using those resources can help us simplify the process of making sense of bibliometric analysis data.

The available tools for bibliometric analysis often have to be used with WoS database, or only one database. In the case of multiple databases being used, the preprocessing is done manually or via code by researchers, only to be visualized in a tool like VosViewer. The software accept different types of files, which also makes the preprocessing more difficult, and makes the analysis more complicated. The use of only one database can be discussed as problematic since we will be restricted to this database for our results. Conversely, Python pandas offers the possibility to build graphs using traditional Excel sheets, and Streamlit enables the visualization of such information in an easy and structured manner.

3. Method

We follow the methodology proposed by [Peffers et al. 2008] in order to perform a DSR to develop a framework to perform bibliographic analysis. This choice is due to the fact that this work focuses on developing an artifact to solve a research problem, which is the main focus of DSR.

As per the steps proposed in [Peffers et al. 2008], we identified the research problem that was explored in section 1, which is the lack of a software for bibliometric analysis that can integrate with any database easily, and that eliminates the need of using multiple tools. Afterwards, we develop an IT artifact that aims to solve this problem. We then test the solution against a generic search on a medical science database to evaluate its performance.

3.1. Architecture

To explain the structure of the tool, a high level architecture was developed. The structure is designed to be simple, allowing scholars to easily understand and, if desired, complement the code with features that better suit their specific needs. The tool uses an Excel with citation metadata that must have columns Author, Title, Abstract, Article Reference, Times Cited, Publication Year and DOI (names must match those exactally).

As illustrated in Figure 1, the main app communicates with two packages: the data and the analysis. In the data package, the user can upload an Excel file and fill in missing citation metadata, choosing whether to fill in all data or a specific field. In the analysis package, the user must have a citation Excel ready with all the metadata they wish to analyze, and they can use this to visualize performance analysis and science mapping metrics. We will discuss these packages in further detail in the following sections.

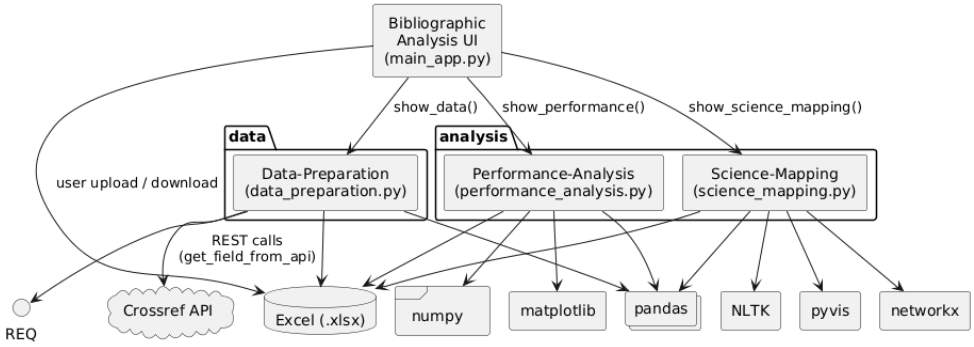

Figure 1. High level architecture

We chose to use Streamlit ([Streamlit 2025]) as a visualization and deployment library, as it is a practical way of sharing applications and creating a visualization of data easily ([Dol and Jawandhiya 2024]). The Crossref API was chosen because it is an open platform with millions of data points available that is updated regularly and can be easily accessible without the need for an API Key ([Crossref 2025]).

The choice of Excel is because one of the challenges described in the previous sections and most software do not cover it, accordingly to the results of [Moral-Muñoz et al. 2020], is the data preparation. By integrating the platform with Excel, we give the possibility for researchers to use all of its resources to keep and edit the data throughout the whole research. The Excel format is easily converted into other formats that are accepted in reference managers, and can be obtained directly through export in many databases, as well as reference managers such as Zotero.

Additionally, by using Python, we can take advantage of the vast number of libraries that are available for this language. In this case, we are using Pandas for data processing, NumPy and Matplotlib for calculation and plotting, NLTK for natural language processing (used in the co-word analysis), pyvis for visualization and networkx for clustering algorithms and calculation of network metrics.

3.2. The data package

One of the main problems we wish to address with this work is providing metrics based on a citation file that could have been obtained from alternative databases, including multiple ones. Currently, not all the databases have options to export the complete set of metadata that is essential to perform bibliometric analysis, such as citation and reference data. For this reason, we opted to implement this data package, which communicates with the Crossref API to fetch missing data.

To use this package, the user needs to enter the app and select the ”Data Preparation” tab and upload an Excel file. The Excel file must have all of the columns specified in Section 3.1. The order of the columns and the completion of the data are not important, as long as the columns are there. The user can then select if they wish to fill all fields or only one field.

The behavior of the algorithm for both of these choices is similar: for each column, the algorithm loops through the entries, identifying the ones that are empty. If the column is DOI, then the algorithm uses the title of the citation to search the Crossref API for an

article DOI with this title. If there is no DOI for this entry or the API does not return any data, then this entry is not filed and the console displays an error message. When the column is Title, then a similar behavior is implemented, except the search term is then DOI instead of title. For every other column, the algorithm tries to use DOI to fetch the data, and if the DOI is missing, it tries to use the title. If both title and DOI are missing or the API cannot fetch any data for this citation, the entry is left empty. This logic is further illustrated in Algorithm 2 as pseudocode.

Algorithm 1 PROCESSEACHFIELD

Require: *required_field*, *data_base*
Ensure: updated *data_base*

$crossref\_field \leftarrow CROSSREF\_AVAILABLE\_FIELDS[required\_field]$
for all rows $\langle index, citation \rangle$ in $data\_base$ **do**
  if $citation[required\_field]$ = "" **then**
    if $crossref\_field = \text{“}DOI\text{”}$ **then**
      $search\_term \leftarrow citation[\text{“}Title\text{”}]$
    else
      $search\_term \leftarrow citation[\text{“}DOI\text{”}]$
    end if
    try
      $field\_value \leftarrow$ GETFIELDFROMAPI$(crossref_field, search_term)$
    if $field\_value \neq$ "" **then**
      $data\_base[index, citation\_field] \leftarrow field\_value$
      $field\_value \leftarrow$ ""
    end if
    catch Exception e
      PRINT(“Unable to get data from API: ”, e)
  end if
end for
return $data\_base$

It is important to highlight that, as mentioned in Figure 2, Title and DOI data are extremely necessary if the user wishes to perform any data extraction. As explained in the previous paragraphs, if none of those are present for a citation, then no data can be retrieved.

For the column reference, a special treatment is required. After querying Crossref for the reference data of a citation, the type of data retrieved varies. Sometimes there is a title and a DOI, sometimes there is only publication year and sometimes there is only one information which Crossref treats as ”unstructured”. This is illustrated in Figure **??**, where we can see that two references contain DOI, while the other reference contains a text in the ”unstructured” key, including miscellaneous information on the article.

To deal with this, we chose to use the title of the reference if present. If the title is not present, then we try to query it using DOI. If the API does not find any data, then the algorithm stores the DOI. Finally, if neither title nor DOI is present, then the algorithm fills the reference with the data inside ”unstructured”. If all of these operations fail, then the

title of the algorithm is filed in a text file called error, which can later be used to calculate the percentage of data that is missing for this database. We can refer to Algorithm 2 for a pseudocode that further illustrates this proccess.

Algorithm 2 PARSE ARICLE REFERENCES

Require: *article*, *references*
Ensure: concatenated reference string *references_line*

$references\_line \leftarrow$ “” ▷ initialise as empty string
for all $reference \in references$ **do**
  $reference\_text \leftarrow$ “”
  if $title \in reference$ **then**
    $reference\_text \leftarrow reference[title] \parallel$ “; “
  else if $doi \in reference$ **then**
    try
      $title \leftarrow$ GETFIELDFROMAPI($“title”, reference[doi]$)
    if $title \neq$ “” **then**
      $reference\_text \leftarrow title \parallel$ “; “
    end if
    catch Exception e
      PRINT(“Could Not Find Title For article: ”, e)
      $reference\_text \leftarrow reference[doi] \parallel$ “; “
  else if “unstructured” $\in reference$ **then**
    $reference\_text \leftarrow reference[“unstructured”] \parallel$ “; “
  else
    PRINT(“Error: ”, e)
    continue
  end if
  $references\_line \leftarrow references\_line \parallel reference\_text$
end for
return $references\_line$

3.3. The analysis package - Performance Analysis

For this part of the algorithm, the user must have the database file with the final data they wish to be used in the performance analysis on an Excel file. This file does not need to be archived using the algorithm described in the previous section, but it must contain all columns present on Section 3.1. Additionally, citation data, author data and publication year data must be present. When a user uploads the Excel, they can visualize performance analysis metrics.

In this section, the user will find a summary of the performance of the field containing the number of unique authors, total citations and average citations. Other metrics are also displayed in this section, such as the h and g indices for authors and the Lorenz curve and Gini coefficient. Graphics such as publications per year and citations per year are also created, to assist researchers in gaining insights about the history of the research field.

Additionally, a graph summarizing the number of articles with missing citations

per year is displayed to assist researchers in elaborating on why information is missing for such files, as displayed in Figure 2. This section can also be used to calculate the margin of error for the results obtained. It is necessary to mention the error data since the data that researchers use can be obtained by API, and thus are not always complete, influencing research results. There can be other reasons why citations are missing for some files, for example, the fact that earlier articles tend not to have citation counts.

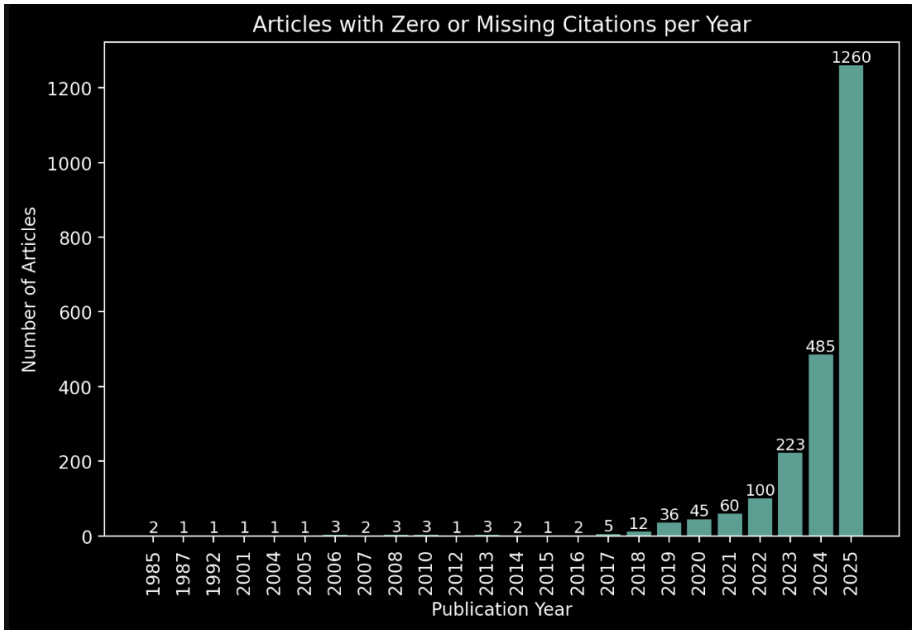

Figure 2. A graph of articles with missing citation data per year shows that most missing citations are in 2025. Recent publications tend to have fewer citation counts.

All of the graphs and calculations use Python's libraries cited in the Architecture Section. To calculate the Lorenz curve, we are following the definition mentioned in the Theoretical Reference: first, we calculate the number of cumulative citations:

$$S_n = \frac{\sum_{i=1}^{k} \text{sorted\_citations}_i}{\sum_{i=1}^{n} \text{sorted\_citations}_i}, \quad k = 1, 2, \ldots, n \tag{7}$$

The y axis of the curve is then the cumulative share of citations. The x axis is the cumulative share of authors in the database, or $1/n$. The matplotlib library allows us to make a continuous plot for these calculations, forming our curve. To calculate the Gini coefficient, we use the discrete approximation described in the Theoretical Reference.

3.4. The analysis package - Science Mapping

Similarly to the previous section, the user must have an Excel file with the final citation metadata, except in this case, it is necessary to have the reference data field for the citations. After uploading this file, the user can observe co-citation clusters, bibliographic coupling clusters and co-word analysis clusters, with a few configuration options.

For the clusters, we chose to display tables to better identify the nodes. The tables are annotated with the chosen network metric, and the user can select clusters separately to

analyze them closely. Note that there is a minimum threshold to consider co-citation pairs or bibliographic-coupling pairs into the clusters, which we chose to be 100. The reasoning is that a smaller number will lead to the creation of clusters that are too small, at the cost of high computational resources.

The algorithm for calculating clusters can be selected by the user between: Greedy, Louvain and Label Propagation. All of the algorithms are calculated using the Networkx library ([Hagberg, Aric et al. 2008]) that implements them as explained in the Theoretical Background section. Further details on the implementation of the algorithms can be found in the library's documentation ([Hagberg, Aric et al. 2008]).

As for the co-word analysis, we opted to display word clouds based on a focus word. This choice is to stimulate scholars to direct their co-word analysis carefully, using this resource to carefully direct their research around keywords they judge important.

To build the word cloud, we use NLTK to filter stop words. Then, we look for subsets containing the focus word and tokenize the resulting lists. The most common words, according to the user selection, will then be put into a Graph using Networkx and then displayed to the user.

In this section, we include information about which percentage of the citation data is missing references, so that the user can include in their research the reasoning why this may occur and argue how this affects their research results.

3.5. Demonstration

To demonstrate the use of this tool, we tested against a test database, which is available in the repository of the tool. This database was extracted from the platform PubMed, a traditional database in the field of medicine. We chose to extract results from a medicine database to demonstrate that this tool can be applied by a researcher in any science discipline. After extracting the database, we used the tool to test its functionalities. We further detail this process in the following sections.

To obtain the database, we went to the PubMed platform and searched for the term "Artificial Intelligence". The search was kept simple and broad for test purposes. We chose to select only the results with open text options and found a total of 87,876 results. To facilitate testing, we extracted the first 10000 results in the PubMed platform to a CSV file and then saved this file in Excel format. The file was modified so that the column names matched the ones from Table 1, and other columns were left intact.

We observe that this database does not allow the extraction of citation and reference data, crucial points to perform bibliometric analysis. We use our implementation to extract this from the Crossref API. Since our database lacks citations and references, we ran the algorithm for those three fields separately. It is important to notice that, because there is a connection with the Crossref API to obtain the data, this needs to be done by a computer connected to the internet and is an operation that can take a long time depending on the size of the database. For our database of 10000 entries, it took more than 2 hours to complete the operation for each field.

We were able to obtain citation data for 7748 articles representing 77.48% of the entries and 9511 of the reference data representing 95.12% of the entries. We were able to see that the citation is well distributed in this database, as the Gini Coefficient is 0.8. This

field has 207877 with 30804. We can also see that most of the articles missing citation data are from 2025, which makes sense since recent work tends to be less cited. This field is relatively new, so it is justified to have a lot of work with few citations.

This database formed 13 co-citation clusters and 23 bibliographic coupling clusters (using the greedy algorithms). The co-citation clusters are displayed in 3. The formation of the clusters happens instantaneously regardless of the size of the database. We are able to see that the co-word cloud identified words like learning, clinical and challenges, associated with the word future in the context of artificial intelligence in the medical field.

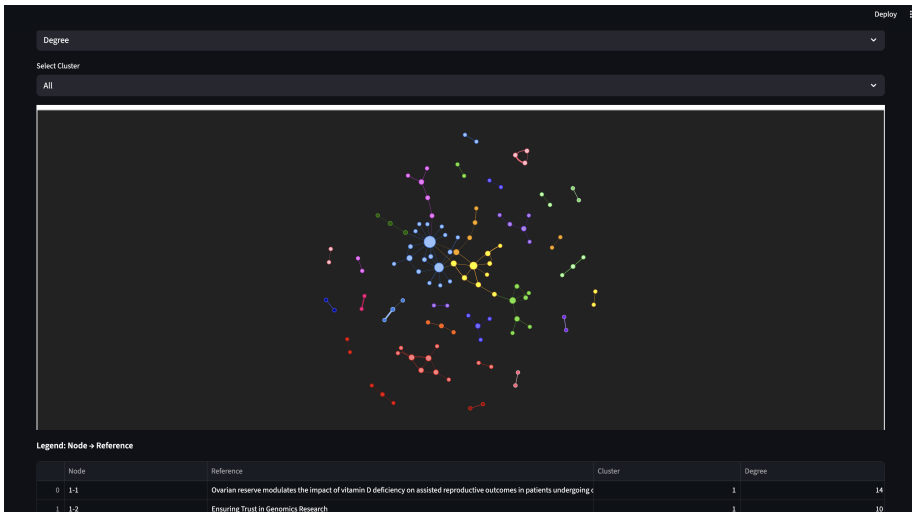

Figure 3. There are 23 biliographic coupling clusters. We can see the dropdown options to select the cluster and the network metric desired.

The demonstration of this tool considered a large dataset, obtained by an alternative database that is not covered by the main solutions found available for bibliographic analysis. We observe that the tool can retrieve the necessary data with a margin of error that is expected, since data is retrieved from API. Additionally, the graphs are visualized instantly without the need for configuration in case of basic use. After loading the database, the user can immediately make observations about the database.

3.6. Adaptation and Reuse

The tool uses Excel format integrating with pandas. This choice is justify by Excel being a popular tool that allows users to personalize and modify fields. We observe that even though Excel is a proprietary tool, the extension .xlsx can be obtained with the use of open tools, for example google sheets.

By presenting this integration with Excel, the tool can assist researchers that do not extract data only from Excel. In case a user requests further options of bibliometric tools to perform analysis, the personalization may be achieved by adding those resources using streamlit.

Additionally, the use of the Crossref API may be changed to other databases depending on the researchers target field and necessities. The data extraction can also be improved in terms of optimization, suiting the needs of users who require to work with larger databases.

The tool accepts as input an Excel file with the exact column names as the ones highlighted here. This factor can be adapted by modifying string names in the $data\_preparation.py$ file, where the column strings are highlighted in the top and can be easily changed.

4. Discussion

Performing bibliometric analysis can provide important insights about the state of the art of a field of knowledge, which is crucial for researchers engaging in relevant research. However, it is not always so simple as it involves many steps: data extraction, preprocessing and visualization, not to mention the selection of which tools to use, and the learning curve that may be required in order to master their features and extract all possible insights from the dataset.

We summarize below the aspects of this tool which contribute to the simplification of this proccess:

- The data extraction is simplified because the practitioner can use this tool to extract data from multiple databases. As the format required is Excel, users can directly extract the citations from the databases using Excel (for example when using SCOPUS) or extract from reference managers such as Zotero that offer this option (when using databases like ACM and IEEE which export to BibTex). To deal with the fact that some databases do not export all data necessary for performing the analysis, researchers can use API already implemented directly on the interface, without any need for programming.
- The preprocessing is facilitated in the sense that the user can use Excel to perform the processing of the data, or export to a citation manager without the need to work with formats like RIS, BIBText and EndNote.
- The practitioner does not need to make themselves familiar with any set of configurations or settings, as the visualization happens immediately after the data upload.
- Customizing is accessible through programming. Due to the vast number of libraries pre-implemented in Python, especially ones that implement statistical and mathematical functionalities, a user who wishes to add more functions can implement graphs without the need to worry about User Interface aspects. The use of Streamlit eliminates the need to implement complex visualization features.

With this implementation, we aim to simplify this process for researchers by making the extraction simpler and minimizing the learning requirements. Additionally, we provide an option that can be used with any database without the need for the researcher to use an API themselves, which can be daunting for practitioners outside the field of computer science.

The solution proposed is thought to be simple, so that the use is intuitive and the features cover all the necessities of a researcher without complex and complicated plugins. Additionally, the source code is available so that it can be adapted for practitioners with more specific needs.

Some limitations of this solution are that the data extraction takes a long time and it depends on the availability of the data in Crossref, which is often missing, as the platform may not have access to all data sources that can be protected.

Another factor is the validation. In this work, we do not perform the validation of the usability of this tool with users of other research fields. The idea of this work is contribute to a simple solution that can be personalization by users with programming knowledge, and that offers basic support for bibliometric analysis tools with the backing of proper algorithms and mathematical tools. The expectation is that researchers can use this as basis for their own research needs, without the ambition of presenting a solution that will satisfy all needs for this community. Future work could build on this foundation to enhance and extend it.

In terms of visualization, the graphs and metrics made available in this implementation are not customizable and do not cover all of the possibilities that one may wish to analyze in a bibliographic analysis. Though we include the metrics that are more commonly cited in the literature, it is impossible to foresee all of the use cases of the bibliometric tools; therefore, specific cases are left out. For simplicity, we also chose to omit complex customization features as they can make the usage experience of the implementation complex, which is something we wish to prevent with this solution.

A future work may therefore involve the addition of more customization options without the need to make the interface of the tool unnecessarily complex. Techniques can be applied to make the data extraction more complete, such as varying the API sources used or taking advantage of some aspects of the databases to facilitate the attainment of data. SCOPUS and WoS offer APIs for data extraction, but the access to these keys is not open and therefore, future work could include obtaining this access in order to improve the coverage of the data that is covered by the API.

Another improvement could be in the time it takes to run the application during the phase of the data extraction. The integration of more complex techniques or with tools like the WoS API could reduce the waiting time of the user.

5. Conclusion

This work proposes a simple tool to facilitate the bibliometric analysis for practitioners, proposing the use of Python and Streamlit to simplify the technical aspects of the review and concentrate on the aspect that matters the most, which is the critical analysis of data and gathering of insights. Through this, we hope that using the tool makes the analysis faster and more comprehensive, so practitioners/researchers may spend more time on other steps of the research.

The perspective of having the results for a bibliographic analysis faster and based on more platforms rather than just WoS or SCOPUS opens up the possibility for this important type of research to be made more often. This increases the possibility that researchers may take this quantitative overview of their research fields on a more frequent basis, enabling a high level comprehension of their research.

Acknowledgement

This work was funded by the 0100_TID4AGRO_4_E project, co-financed by the European Regional Development Fund (ERDF), managed through the INTERREG VI-A Spain-Portugal (POCTEP) Programme 2021-2027 of the European Commission. It has been co-financed 85% by the European Union, the European Regional Development Fund and

the Regional Government of Extremadura. Managing Authority: Ministry of Finance. Grant: GR24099. It is also funded by I+D+i PID2024-156158OB-I00 project, financed by MICIU/AEI/ 10.13039/501100011033/ FEDER, UE.

This study was also funded by the National Council for Scientific and Technological Development (CNPq) for the support received for the research project under CNPq/MCT Call No. 403677/2025-4 and by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

Alskaf, E., Dutta, U., Scannell, C. M., and Chiribiri, A. (2022). Deep learning applications in coronary anatomy imaging: A systematic review and meta-analysis. *Journal of Medical Artificial Intelligence*, 5(11):1–11.

Barbon Jr, S., Tavares, G. M., and Kido, G. (2017). Artificial and natural topic detection in online social networks. *iSys - Revista Brasileira de Sistemas de Informacao*, 10(1):80–98.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008.

Brandes, U. (2001). A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(2):163–177.

Clauset, A., Newman, M. E. J., and Moore, C. (2004). Finding community structure in very large networks. *Physical Review E*, 70(6):066111.

Cordasco, G. and Gargano, L. (2011). Community detection via semi-synchronous label propagation algorithms. *arXiv preprint arXiv:1103.4550*. Journal reference: Int. J. of Social Network Mining, 2012, Vol. 1, No. 1, pp. 3–26.

Costa, A. P., Moresi, E. A. D., Pinho, I., and Halaweh, M. (2023). Integrating bibliometrics and qualitative content analysis for conducting a literature review. In *2023 24th International Arab Conference on Information Technology (ACIT)*, pages 1–8, Ajman, United Arab Emirates. IEEE.

Crossref (2025). Crossref rest api. `https://api.crossref.org/`. Accessed: 2025-09-22.

de Sousa Araújo, G., Santana, E. E. C., Júnior, A. F. L. J., and Lobato, F. M. F. (2025). The artificial intelligence integration in the brazilian legal sector: A systematic review. In *Anais do XXI Simpósio Brasileiro de Sistemas de Informação (SBSI 2025)*, pages 575–584, Porto Alegre. Sociedade Brasileira de Computação.

Ding, Y. (2009). Pagerank for ranking authors in co-citation networks. *Journal of the American Society for Information Science and Technology*, 60(11):2229–2243.

Dol, S. M. and Jawandhiya, P. M. (2024). Data visualization for the dataset collected from education sector using python. In *2024 1st International Conference on Communications and Computer Science (InCCCS)*, pages 1–6.

Donthu, N., Kumar, S., Mukherjee, D., Pandey, N., and Lim, W. M. (2021). How to conduct a bibliometric analysis: An overview and guidelines. *Journal of Business Research*, 133:285–296.

Dorfman, R. (1979). A formula for the gini coefficient. *The Review of Economics and Statistics*, 61(1):146–149.

Ellegaard, O. and Wallin, J. A. (2015). The bibliometric analysis of scholarly production: How great is the impact? *Scientometrics*, 105(3):1809–1831.

Farris, F. A. (2010). The gini index and measures of inequality. *The American Mathematical Monthly*, 117(10):851–864.

Gastwirth, J. L. (1971). A general definition of the lorenz curve. *Econometrica*, 39(6):1037–1039.

Graeff, C. A., Farias, K., and Carbonera, C. E. (2023). On the prediction of software merge conflicts: A systematic review and meta-analysis. In *Proceedings of the SBSI '23: XIX Brazilian Symposium on Information Systems*.

Guleria, H. V., Luqmani, A. M., Deo, S., Devendra, K. H., Sharma, K., Mishra, S., Bidwe, R. V., Zope, B., and Buchade, A. (2023). 'big news' morgans: A chatbot for f1 news summarization. In *2023 International Conference on Integration of Computational Intelligent Systems (ICICIS)*, pages 1–6. IEEE.

Hagberg, Aric, Schult, Daniel, and Swart, Pieter (2008). *NetworkX: Network Analysis in Python*. Software available at `https://networkx.org/`.

Hakimi, S. L. (1964). Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations Research*, 12(3):450–459.

Ikotun, A. M., Ezugwu, A. E., Abualigah, L., Abuhaija, B., and Jia, H. (2023). K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Information Sciences*, 622:178–210.

Kheddar, H. (2025). Transformers and large language models for efficient intrusion detection systems: A comprehensive survey. *Information Fusion*, 124:103347.

Koçak, M. and Akçalı, Z. (2025). The published role of artificial intelligence in drug discovery and development: a bibliometric and social network analysis from 1990 to 2023. *Journal of Cheminformatics*, 17(1):71.

Kudo, T. N., Bulcão-Neto, R. F., Vincenzi, A. M. R., Souza, É. F. D., and Felizardo, K. R. (2022). Using evidence from systematic studies to guide a phd research in requirements

engineering: An experience report. *Journal of Software Engineering Research and Development*, 10(7):1–12.

Laurett, N. S. and Ribeiro, F. N. (2022). Caracterização das publicações e relações entre mídias alternativas polarizadas no facebook. In *Anais do Brazilian Workshop on Social Network Analysis and Mining (BraSNAM)*, pages 133–144.

Lemos, L. C., Ralha, C. G., Claro, D. B., Maciel, R. S. P., Argolo, A. A., and Linhares, C. D. G. (2024). A temporal network visualization and data analysis on two decades of sbsi. In *Proceedings of the XX Brazilian Symposium on Information Systems (SBSI 2024)*, pages 1–12. Association for Computing Machinery.

Maz-Machado, A., Torralbo-Rodríguez, M., Vallejo-Ruíz, M., and Bracho-López, R. (2010). Análisis bibliométrico de la producción científica de la universidad de málaga en el social sciences citation index (1998-2007). *Revista Española de Documentación Científica*, 33(4):582–599.

Mioto, V. and Vignatti, A. L. (2025). Beyond boundaries: Collaboration networks and research output in brazilian computer science. In *Brazilian Workshop on Social Network Analysis and Mining (BRA/SNAM)*, Curitiba, PR, Brazil.

Modak, N. M., Beydoun, G., Merigó, J. M., Rahimi, I., and Susilo, W. (2025). 40 years of computer standards & interfaces: A bibliometric retrospective. *Computer Standards & Interfaces*, 95:104046.

Monath, N., Dubey, A., Guruganesh, G., Zaheer, M., Ahmed, A., McCallum, A., Mergen, G., Najork, M., Terzihan, M., Tjanaka, B., Wang, Y., and Wu, Y. (2021). Scalable hierarchical agglomerative clustering. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD '21)*, pages 1245–1255. ACM.

Moral-Muñoz, J. A., Herrera-Viedma, E., Santisteban-Espejo, A., and Cobo, M. J. (2020). Software tools for conducting bibliometric analysis in science: An up-to-date review. *El profesional de la información*, 29(1):e290103.

Muhammad Qadir, H., Suleman, M. T., Khan, R. A., Sohaib, M., Hasan, M. J., and Hussain, S. A. (2025). Optimizing learning outcomes: a deep dive into hybrid ai models for adaptive educational feedback. *Journal of Big Data*, 12(1):144.

Neely, A. (2005). The evolution of performance measurement research: Developments in the last decade and a research agenda for the next. *International Journal of Operations & Production Management*, 25(12):1264–1277.

Negre, C. F. A., Morzan, U. N., Hendrickson, H. P., Pal, R., Lisi, G. P., Loria, J. P., Rivalta, I., Ho, J., and Batista, V. S. (2018). Eigenvector centrality for characterization of protein allosteric pathways. *Proceedings of the National Academy of Sciences*, 115(52):E12201–E12208.

Nielsen, F. (2016). *Hierarchical Clustering*, pages 195–211. Springer.

Nolêto, R. M. A., Nolêto, C., Santos, N. P. S., and Madeira, A. M. A. (2023). Inovações no reconhecimento e detecção de animais: Uma análise da literatura com Ênfase em redes neurais e aprendizado de máquina. In *16º Encontro Unificado de Computação do Piauí (ENUCOMPI)*, pages 33–40, Piripiri, PI, Brasil. Sociedade Brasileira de Computação.

Pan, W., Jian, L., and Liu, T. (2019). Grey system theory trends from 1991 to 2018: a bibliometric analysis and visualization. *Scientometrics*, 121(3):1407–1434.

Passas, I. (2024). Bibliometric analysis: The main steps. *Encyclopedia*, 4(2):1014–1025.

Peffers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S. (2008). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77.

Soares, R. H. S., Fernandes, J. H. C., and Sampaio, R. (2019). Formal information flows among top authorities of the brazilian federal government based on co-word analysis of data published in the official gazette. In *Anais do Brazilian Workshop on Social Network Analysis and Mining*, pages 1–6. Sociedade Brasileira de Computação.

Streamlit (2025). Streamlit. `https://streamlit.io/`. Accessed: 2025-09-22.

Tenório, K., Santos, J., Accete, V., Remigio, S., da Silva, A. P., Dermeval, D., Bittencourt, I. I., and Marques, L. B. (2021). On the joint use of artificial intelligence and brain-imaging techniques in technology-enhanced learning environments: A systematic literature review. *Revista Brasileira de Informática na Educação (RBIE)*, 29:502–518.

Uzeda, L. E., Parreiras, M., and Xexéo, G. (2023). Exploring the intersection of game-based learning and sustainable education in engineering: A bibliometric analysis. In *Anais Estendidos do XXII Simpósio Brasileiro de Jogos e Entretenimento Digital (SBGames)*, pages 683–694, Rio Grande/RS, Brasil. Sociedade Brasileira de Computação (SBC).

van Raan, A. (2014). Advances in bibliometric analysis: research performance assessment and science mapping. In *Research Performance Assessment and Science Mapping*. Portland Press Limited. Disponível em: `https://www.cwts.nl/TvR/documents/TonvR%282%29.pdf`.